



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/080,928	02/25/2002	Thomas J. Martin	002854-39	7788

22204 7590 03/18/2005

NIXON PEABODY, LLP
401 9TH STREET, NW
SUITE 900
WASHINGTON, DC 20004-2128

EXAMINER

COURTENAY III, ST JOHN

ART UNIT PAPER NUMBER

2126

DATE MAILED: 03/18/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/080,928

Applicant(s)

MARTIN ET AL.

Examiner

St. John Courtenay III

Art Unit

2126

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 25 February 2002.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-25 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-25 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.


Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 25 February 2002 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.


ST. JOHN COURTENAY III
PRIMARY EXAMINER

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date April 3, 2002.

- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

Application/Control Number:
10/080,928
Art Unit: 2126

Page 2

Detailed Action

The preliminary amendment, filed May 22, 2002, amended originally filed claims 1, 15, 21, and 23, and added new claims 24 and 25. Claims 1-25 are now pending for examination on the merits.

Applicant's claim for priority under 35 U.S.C. § 119(e) with respect to provisional applications 60/350,351, filed Jan. 24, 2002 and 60/354,235, filed on Feb. 6, 2002, is acknowledged.

With respect to independent claim 1 any terminology in the preamble that limits the structure of the claimed invention must be treated as a claim limitation. See, e.g., Corning Glass Works v. Sumitomo Elec. U.S.A., Inc., 868 F.2d 1251, 1257, 9 USPQ2d 1962, 1966 (Fed. Cir. 1989); Pac-Tec Inc. v. Amerace Corp., 903 F.2d 796, 801, 14 USPQ2d 1871, 1876 (Fed. Cir. 1990). See also In re Stencel, 828 F.2d 751, 4 USPQ2d 1071 (Fed. Cir. 1987).

Accordingly, the "computer architecture" recited in the preamble of independent claim 1 is considered by the Examiner as limiting the structure of the claimed invention to provide hardware computing elements required for execution of the claimed plural software application programs.

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

However, the language of independent claims 15 and 18 raises a question as to whether the aforementioned claims are directed merely to an abstract idea that is not tied to a technological art, environment or machine which would result in a practical application producing a useful, concrete, and tangible result to form the basis of statutory subject matter under 35 U.S.C. 101.

Independent claims 15 and 18 do not appear to require any computer hardware to implement the claimed invention. Claim 15 merely recites a "method of defining." Likewise, claim 18 recites an object "definition." These claims appear to define the metes and bounds of an invention comprised of software alone. Software alone, without a machine, is incapable of transforming any physical subject matter by chemical, electrical, or mechanical acts.

If the "acts" of a claimed process manipulate only numbers, abstract concepts or ideas, or signals representing any of the foregoing, the acts are not being applied to appropriate subject matter. In re Schrader, 22 F.3d 290 at 294-95, 30 USPQ2d 1455 at 1458-59 (Fed. Cir. 1994).

Transformation of data by a machine constitutes statutory subject matter if the claimed invention as a whole accomplishes a practical application. That is, it must produce a "useful, concrete and tangible result." State Street, 149 F.3d 1368, 1373, 47 USPQ2d 1596 at 1600-02 (Fed. Cir. 1998). MPEP 2106.

State Street required transformation of data by a machine before it applied the "useful, concrete, and tangible test." However,

State Street does not hold that a "useful, concrete and tangible result" alone, without a machine, is sufficient for statutory subject matter. State Street, 149 F.3d at 1373, 47 USPQ2d at 1601.

Claims 15-23 are rejected under 35 U.S.C. 101 because the claimed invention, appearing to be comprised of software alone without claiming associated computer hardware required for execution, is not supported by either a specific and substantial asserted utility (i.e., transformation of data) or a well established utility (i.e., a practical application).

35 U.S.C. § 112, 1st paragraph

The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 15-23 are also rejected under 35 U.S.C. 112, first paragraph. Specifically, since the claimed invention is not supported by either a specific and substantial asserted utility or a well established utility for the reasons set forth above, one skilled in the art would not know how to use the claimed invention.

35 U.S.C. § 112, 2nd paragraph

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

As per claims 15-23:

Claims 15-23 are rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting essential elements, such omission amounting to a gap between the elements. See

MPEP § 2172.01. The omitted elements are computer hardware necessary to execute the claimed software and render the invention operative.

As per claim 24:

Dependent claim 24 is rejected under 35 U.S.C. 112, second paragraph because claim 24 recites the limitation "corporate architecture" on page 4 of the preliminary amendment received May 22, 2002. There is insufficient antecedent basis for this limitation in the claim. Specifically, a "computer architecture" is recited in claim 1 from which claim 24 depends.

35 U.S.C. § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1- 25 are rejected under 35 U.S.C. § 102(b) as being anticipated by **Schmitter** (U.S. Patent 5,583,983).

As per independent claim 1:

Schmitter teaches a computer architecture for sharing information between plural applications having disparate data structures, the architecture comprising:

- plural applications, at least one of the applications having a data structure that is different from another of the

applications [e.g., see col. 2, line 54, "object oriented applications" and see first and second computing systems and associated discussion col. 2, lines 59, 67];

- an application integration platform including logic for exchanging information between the plural applications [e.g., see "In order to provide a system for the development of object-oriented applications and for cross-platform deployment, or further development, of object-oriented applications, a library of objects, defined using a standard high-level language, is provided in a form that is independent of any single operating system. The standard library of objects is provided to a selected computer system having a first operating system and an object-oriented development environment. Within the object-oriented development environment, the software engineer assembles an application on the basis of a problem specification. When the selected objects have been linked in the desired manner, an archive, or object document, of the application is produced. The object document is provided in a portable form that can then be provided to a second computing system having a different native operating system than the first computing system upon which the object-oriented development environment is installed." and associated discussion col. 2, beginning line 53] ; and
- at least one common object definition specifying common objects to be used for exchanging data between the applications and including a canonical object defining elements of a standard object that are common between data structures of at least any two of the plural applications, the common object further including at least one extension defining application specific, industry specific, or user specific elements, the canonical object being exposed to all of the applications through the application integration platform, the extension being exposed only to selected ones of the plural applications [e.g., see " The canonical object definitions should be based upon native objects of each type that have the greatest number of attributes, so that the canonical definitions will be comprehensive enough to permit archiving of all native objects of each type, regardless of the number of attributes that are required for each type of object in any particular environment. Of course, it is to be expected that objects of a given type that are developed after the establishment of the canonical definition for that type, may have a greater number of attributes than is permitted by the canonical definition. "] and associated

discussion col. 13, beginning line 61] .

As per independent claim 15:

Schmitter teaches a method of defining a common data object for sharing information between plural applications having disparate data structures, the method comprising:

- identifying one or more primary applications each having a data structure [e.g., see col. 2, line 54, "object oriented applications" and see first and second computing systems and associated discussion col. 2, lines 59, 67];
- determining common data elements between at least any two of the data structures [e.g., see "As described hereinabove, the portability of an object-oriented structure to a particular target environment would require that the library contain definitions of the objects therein that are compatible with the target environment. Thus, a large number of conditional object definitions would be required in order to provide a single, universal object library that would be widely and generally applicable for the development of various object-oriented structures. " and associated discussion col. 10, beginning line 8];
- selecting elements of a canonical object that correspond to the common elements [e.g., see "One way to accomplish cross-platform portability of a native object library is to establish, for each object in the library, a procedure by which each object can be archived in a canonical form." and associated discussion col. 10, beginning line 20];
- adjusting the canonical object based on a common object standard [e.g., see "development of portable structure" and associated discussion col. 10, beginning line 30]; and
- adding at least one application specific, industry specific, or user specific extension to the data elements of the canonical

object [e.g., see "The archiver responds to the attributes transmitted from the "putTo:" procedures by entering the transmitted attributes into an archive according to a predetermined canonical format for each type of object. In order to translate various native object references into the predetermined canonical format, the archiver can be conditionally coded so that it translates each native reference to a canonical reference within the environment in which the archiver is compiled." and associated discussion col. 10, beginning line 33].

As per independent claim 18:

Schmitter teaches a common object definition for common objects used for sharing information between plural applications having disparate data structures, the definition comprising:

- a canonical object defining elements of a standard object that are common between data structures of the plural applications [e.g., see "The canonical object definitions should be based upon native objects of each type that have the greatest number of attributes, so that the canonical definitions will be comprehensive enough to permit archiving of all native objects of each type, regardless of the number of attributes that are required for each type of object in any particular environment. Of course, it is to be expected that objects of a given type that are developed after the establishment of the canonical definition for that type, may have a greater number of attributes than is permitted by the canonical definition. "]; and
- at least one extension defining application specific or user specific elements, the canonical object being exposed to all of the applications, the extension being exposed only to selected ones of the plural applications [e.g., see "The archiver responds to the attributes transmitted from the "putTo:" procedures by entering the transmitted attributes into an archive according to a predetermined canonical format for each type of object. In order to translate various native object references into the predetermined canonical format, the archiver can be conditionally coded so that it translates each native reference to a canonical reference within the environment in which the archiver is compiled." and associated discussion col. 10, beginning line 33].

As per dependent claim 2:

Schmitter teaches the at least one extension comprises an application specific extension having data elements used only by a first of the plural applications and a user specific extension having data elements not in the canonical object but desired by a specific user [see e.g., "Of course, the archiver may be more specifically arranged to translate native object references in one environment into native object reference pertaining to another specific environment in an alternative embodiment for providing one-way portability between only two environments." col. 10, beginning line 52].

As per dependent claim 3:

Schmitter teaches the common object definition comprises a tree like structure [see "translation table" and associated discussion col. 11, beginning line 27].

As per dependent claim 4:

Schmitter teaches each of the canonical object and the extensions are represented by a separate node in the common object definition [see "translation table" and associated discussion col. 11, beginning line 27; see also native object and canonical definition discussion col. 14, beginning line 4].

As per dependent claim 5:

Schmitter teaches each of the canonical object and the extensions are represented by a distinct DTD in the common object definition [see e.g., "The inclusion of conditional, native object definitions within the universal object library 11 allows the appearance of the user interface of an application to assume the standard appearance of a user interface within each computer system upon which the application is deployed." col. 8, line 31; see also "object document 28" and associated discussion, col. 9, line 18].

As per dependent claim 6:

Schmitter teaches the common object definition references another common object definition [see e.g., "The main routine of the execution program 34 is compiled and linked with the library prior to execution, so that it

may access any objects required by the application." col. 9, discussion beginning line 23].

As per dependent claim 7:

Schmitter teaches means for cross referencing data elements in the common object definition with corresponding data elements in the applications [see e.g., "The canonical object definitions should be based upon native objects of each type that have the greatest number of attributes, so that the canonical definitions will be comprehensive enough to permit archiving of all native objects of each type, regardless of the number of attributes that are required for each type of object in any particular environment. Of course, it is to be expected that objects of a given type that are developed after the establishment of the canonical definition for that type, may have a greater number of attributes than is permitted by the canonical definition." col. 13, discussion beginning line 61].

As per dependent claim 8:

Schmitter teaches the application integration platform is operative to enforce plural system of record policies [see e.g., "One approach to archiving a native object which has more attributes than its canonical definition is to provide the object with the ability to enter "custom" information into the object document in such a way that the custom information will subsequently be recognized as such." col. 14, discussion beginning line 4].

As per dependent claim 9:

Schmitter teaches the system of record policies include a federated policy in which different ones of the applications is responsible for updating different portions of common business objects corresponding to a particular common business object definition [see e.g., "In the proliferation of object-oriented programming techniques, it is expected that there will be some environments which provide a greater variety of objects than other environments. Thus, it may occur in some situations that there exist more established canonical types of objects than are supported within a particular computing system." col. 15, discussion beginning line 25].

As per dependent claim 10:

Schmitter teaches the system of record policies include a revolving policy in which different ones of the applications is responsible for updating common business objects corresponding

to a particular common business object definition at different points of the life cycle of the common business object [see e.g., "More than one archive may be associated with an object-oriented application. Objects within an application, for example, may have attributes which include other previously-developed, object-oriented structures having their own object documents." col. 15, discussion beginning line 61].

As per dependent claims 11 & 12:

Schmitter teaches the system of record policies include a rules based policy in which common business objects corresponding to a particular common business object definition are updated in different manners based on external factors applied to predetermined rules [see e.g., "More than one archive may be associated with an object-oriented application. Objects within an application, for example, may have attributes which include other previously-developed, object-oriented structures having their own object documents." col. 15, discussion beginning line 61].

As per dependent claims 13 & 14:

Schmitter teaches the integration platform comprises at least one connector having a transformation map, the transformation map comprising plural map modules, as claimed [see e.g., "The collection of association objects would be defined by a master object document. The master object document would preferably provide definitions of association objects for allowing the archiver and de-archiver respectively to set and to get the attributes of each native object in the environment, and to appropriately translate canonical and native references. Then, deployment within new environments, or deployment within a newly-modified environment, would only require appropriate modification of the master object document defining the association objects for an intended target environment", col. 17, discussion beginning line 2].

As per dependent claims 16 & 17:

Schmitter teaches the adding step comprises adding data elements of a specified application to maintain functionality of the specified application in a system using the common object, as claimed [e.g., see "The archiver responds to the attributes transmitted from the

"putTo:" procedures by entering the transmitted attributes into an archive according to a predetermined canonical format for each type of object. In order to translate various native object references into the predetermined canonical format, the archiver can be conditionally coded so that it translates each native reference to a canonical reference within the environment in which the archiver is compiled." and associated discussion col. 10, beginning line 33].

As per dependent claim 19:

See the rejection of claim 2 above.

As per dependent claim 20:

See the rejection of claim 3 above.

As per dependent claim 21:

See the rejection of claim 4 above.

As per dependent claim 22:

See the rejection of claim 5 above.

As per dependent claim 23:

See the rejection of claim 6 above.

As per dependent claims 24 & 25:

Schmitter teaches at least one extension comprise an industry specific extension having data elements adapted to a specific industry and the adding step comprises adding data elements adapted to a specific industry, as claimed [see e.g., "Of course, the archiver may be more specifically arranged to translate native object references in one environment into native object reference pertaining to another specific environment in an alternative embodiment for providing one-way portability between only two environments." col. 10, beginning line 52].

Prior Art not relied upon:

Please refer to the references listed on the attached PTO-892 which are not relied upon in the claim rejections detailed above.

Application/Control Number:
10/080,928
Art Unit: 2126

Page 13

How to Contact the Examiner:

Any inquiry concerning this communication or earlier communications from the examiner should be directed to St. John Courtenay III, whose telephone number is 571-272-3761. A voice mail service is also available at this number. The Examiner can normally be reached on Monday - Friday, 8:00 AM - 4:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, An Meng-AI who can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

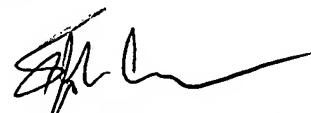
Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

All responses sent by U.S. Mail should be mailed to:

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

**PTO CENTRAL FAX NUMBER:
703-872-9306**

- Any inquiry of a general nature or relating to the status of this application should be directed to the **TC 2100 Group receptionist: (571) 272-2100.**


ST. JOHN COURTENAY III
PRIMARY EXAMINER